

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29

What is claimed is:

1. A method for run-time configurable caching of component factories, comprising:

launching an object-oriented application;

creating and initializing a plurality of cacheable factory objects;

creating and initializing a plurality of cache objects;

assigning one or more cache objects of the plurality of cache objects to one or more cacheable factory objects of the plurality of cacheable factory objects;
and

manipulating one or more objects contained in the plurality of cache objects.
2. The method of claim 1, wherein the plurality of cache objects contain a plurality of methods to add an object to a cache, remove an object from the cache, and find an object in the cache.
3. The method of claim 1, wherein the plurality of cacheable factory objects contain a plurality of methods to get an object from a cache, and to couple a cache object to a cacheable factory object.
4. The method of claim 1, wherein the plurality of cacheable factory objects derive from a corresponding plurality of factory objects.

1 5. The method of claim 4, wherein the plurality of factory objects contain a
2 plurality of methods to create an object, obtain an object identifier, and get a
3 database connection object.

4
5 6. The method of claim 4, wherein the plurality of factory objects and the
6 plurality of cache objects derive from a common base object.

7
8 7. The method of claim 1, further comprising:

9
10 a plurality of cache statistics objects; and

11
12 a plurality of cache configuration objects.

13
14 8. The method of claim 7, wherein the plurality of cache statistics objects
15 contain a plurality of methods to determine the number of cache accesses,
16 the number of times a cache access returned an empty result, the size of a
17 cache, and a reset command.

18
19 9. The method of claim 7, wherein the plurality of cache configuration
20 objects contain a plurality of methods to empty a cache, set and get a
21 maximum cache size, and set and get the cache type.

22
23 10. The method of claim 1, wherein initializing the plurality of cache
24 objects, further comprises:

25
26 setting a cache type for each cache object of the plurality of cache
27 objects; and

28

1 setting a maximum size for the number of objects contained in each
2 cache object of the plurality of cache objects.

3

4 11. The method of claim 10, wherein the cache type for each cache object
5 of the plurality of cache objects determines how each object in the cache is
6 removed from the cache.

7

8 12. The method of claim 1, wherein the application interacts with the
9 plurality of cacheable factory objects to manipulate the one or more objects
10 located in the plurality of cache objects.

11

12 13. The method of claim 12, wherein manipulating the one or more objects
13 further comprises adding one or more objects to the plurality of cache objects.

14

15 14. The method of claim 13, wherein adding the one or more objects
16 located in the plurality of cache objects, further comprises:

17

18 the application sending a message to a cacheable factory object of the
19 plurality of cacheable factory objects to add the one or more objects to the
20 cache object coupled to the cacheable factory object;

21

22 the cacheable factory object receiving the message and sending a message
23 to the cache object to add the one or more objects to the cache coupled to the
24 cache object.

25

26 15. The method of claim 12, wherein manipulating the one or more objects
27 further comprises removing one or more objects from the plurality of cache
28 objects.

29

1 16. The method of claim 15, wherein removing the one or more objects
2 located in the plurality of cache objects, further comprises:

3
4 the application sending a message to a cacheable factory object of the
5 plurality of cacheable factory objects to remove the one or more objects
6 located in a cache object coupled to the cacheable factory object;

7
8 the cacheable factory object receiving the message and sending a message
9 to the cache object to remove the one or more objects from the cache coupled
10 to the cache object.

11
12 17. The method of claim 16, wherein initializing the plurality of cache
13 objects further comprises:

14
15 setting a cache type for each cache object of the plurality of cache
16 objects; and

17
18 setting a maximum size for the number of objects contained in each
19 cache object of the plurality of cache objects.

20
21 18. The method of claim 17, wherein the cache type for each cache object
22 of the plurality of cache objects determines how each object in the cache is
23 removed from the cache.

24
25 19. The method of claim 12, wherein manipulating the one or more objects
26 further comprises locating one or more objects from the plurality of cache
27 objects.

20. The method of claim 19, wherein locating the one or more objects located in the plurality of cache objects, further comprises:

the application sending a message to a cacheable factory object of the plurality of cacheable factory objects to locate the one or more objects located in a cache object coupled to the cacheable factory object;

the cacheable factory object sending a message to the cache object determining whether any of the one or more objects are contained in the cache object;

if able to locate the one or more objects, the cache object returning any of the one or more objects contained in the cache object; and

if unable to locate the one or more objects, the cacheable factory object accessing the one or more objects from a database, and adding the one or more objects to the cache object.

21. The method of claim 12, wherein manipulating the one or more objects further comprises identifying which of the one or more objects contained in the plurality of cache objects are not the same as the corresponding one or more objects contained in a database.

22. A structure for run-time configurable caching of component factories, comprising:

one or more factory objects, coupled to an object-oriented application;

1 one or more cacheable factory objects, derived from the one or more
2 factory objects; and

3
4 one or more cache objects, coupled to the cacheable factory objects.

5
6 23. The structure of claim 22, wherein the object-oriented application
7 interacts with the plurality of cacheable factory objects in order to manipulate
8 one or more objects contained in the plurality of cache objects.

9
10 24. The structure of claim 22, wherein a plurality of objects contained in the
11 one or more cache objects can be uniquely identified.

12
13 25. The structure of claim 22, wherein the plurality of cache objects contain
14 a plurality of methods to add an object to a cache, remove an object from the
15 cache, and find an object in the cache.

16
17 26. The structure of claim 22, wherein the plurality of cacheable factory
18 objects contain a plurality of methods to get an object from a cache, and to
19 couple a cache object to a cacheable factory object.

20
21 27. The structure of claim 22, wherein the plurality of cacheable factory
22 objects derive from a corresponding plurality of factory objects.

23
24 28. The structure of claim 22, wherein the plurality of factory objects
25 contain a plurality of methods to create an object, obtain an object identifier,
26 and get a database connection object.

27
28 29. The structure of claim 22, wherein the plurality of factory objects and
29 the plurality of cache objects derive from a common base object.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20

30. The structure of claim 22, further comprising:

a plurality of cache statistics objects; and

a plurality of cache configuration objects.

31. The structure of claim 30, wherein the plurality of cache statistics objects contain a plurality of methods to determine the number of cache accesses, the number of times a cache access returned an empty result, the size of a cache, and a reset command.

32. The structure of claim 30, wherein the plurality of cache configuration objects contain a plurality of methods to empty a cache, set and get a maximum cache size, and set and get the cache type.

33. The structure of claim 30, wherein the plurality of cache statistics objects and the plurality of cache configuration objects derive from the common base object.